U.S. Express Mail No.: EU245163358US
Filing Date: June 30, 2003

PATENT
Docket No. CX03005USU (02CXT0078D)

SYSTEM FOR DATA ENCRYPTION AND DECRYPTION OF DIGITAL DATA ENTERING AND LEAVING MEMORY

INVENTOR

WINEFRED WASHINGTON

BACKGROUND OF THE INVENTION

[001] **1. Field of the Invention.**

[002] This invention relates generally to data encryption and more particularly to data encryption of digital data in memory.

[003] **2. Related Art.**

[004] Unauthorized copying of digital content is a growing problem with the proliferation of digital media. Digital media is being delivered to consumers via digital cable systems, digital satellites systems, telephone systems, data networks, and storage devices such as CDs, DVDs, VCDs, etc.... Examples of such a system in operation includes a cable system delivering encrypted digital video to a subscriber's set-top box. In attempts to stop the unauthorized copying of digital media, companies and industries have developed digital encryption methods and systems.

[005] The digital encryption methods and systems encrypt the digital data being delivered to a digital device and provide data encryption keys. In other digital devices, the data encryption keys are secret hard-wired keys or programmable fuses. The encrypted digital data is then decrypted employing the data encryption keys stored in the memory of the digital device when accessed by a user. Unsecure rewriteable memory or hard-wired keys in the digital device enables industrious hackers to monitor accesses to the memory, reverse engineer data encryption

keys and access the digital data, or copy the unencrypted data from unsecure rewriteable memory.

[006] Therefore, there is a need for methods and systems for encrypting and decrypting digital data when delivered to a user that overcomes the disadvantages set forth above and others previously experienced.

SUMMARY

[007] Methods and systems consistent with the present invention provide data encryption and decryption at the memory interface in a digital device. A clock generator and a linear feedback shift register in the digital device generates one or more encryption keys for use by an encryption algorithm every time the digital device is cycled or reset. An inaccurate clock is employed that oscillates at different frequencies depending on gate delays, temperature and voltage. It is therefore the clock rate that is unpredictable and is very likely to differ upon cycling or resetting the digital device resulting in different keys being generated. Further, a pseudo random number may be generated for use by other programs on the digital device or for secure transactions.

[008] Other systems, methods, features and advantages of the invention will be or will become apparent to one with skill in the art upon examination of the following figures and detailed description. It is intended that all such additional systems, methods, features and advantages be included within this description, be within the scope of the invention, and be protected by the accompanying claims.

BRIEF DESCRIPTION OF THE FIGURES

[009] The components in the figures are not necessarily to scale, emphasis instead being placed upon illustrating the principles of the invention. In the figures, like reference numerals designate corresponding parts throughout the different views.

[010] FIG. 1 illustrates a block diagram of an exemplar digital device with an encryption circuit.

[011] FIG. 2 illustrates a block diagram of the encryption circuit of FIG. 1.

[012] FIG. 3 illustrates a block diagram of the encryption key generator and key store of FIG. 2.

[013] FIG. 4 illustrates a circuit diagram of the clock generation block of FIG. 3.

[014] FIG. 5 is a circuit with a zero primer circuit employed with the linear shift register of FIG. 3.

[015] FIG. 6 is a flow diagram of the steps of digital data encryption of digital data entering memory.

[016] FIG. 7 is a flow diagram of the steps of retrieving and decryption of digital data contained in memory.

DETAILED DESCRIPTION

[017] Turning first to FIG 1, that figure shows a block diagram of an exemplar digital device 100 with an encryption circuit 102. The digital device 100 has a memory controller 104, data buffer 106, input/output (I/O) register 108, I/O register 110, control pad 112 and control pad114. The encryption circuit 102 also has a control block 118 and data I/O block 116. The

digital device 100 may be a digital set-top box, MP-3 player, encrypted CD/DVD/VCD player, or digital video recorder to name but a few example digital devices.

[018] The memory controller 104 communicates with the encryption circuit 102 and I/O register 110. The encryption circuit 102 communicates with the memory controller 104, data buffer 106 and data register 108. Both I/O register 108 and I/O register 110 communicate with respective control pads 112 and 114. A control pad is an interface point to another circuit or bus, such as a pin on an integrated circuit package, a solder point pad on circuit board, a doped connection within an integrated circuit, and a wire for connection to another circuit.

[019] The encryption circuit 102 is placed between the data buffer 106 and I/O register 108. The encryption circuit 102 employs the memory control signal from the memory controller 104 to extract the row and bank information employed to read and write digital data to memory. In the present example, there are no latencies or clock offsets inserted in the data paths. In other implementations, clock cycles or other latencies may be designed into the encryption circuit 102. The encryption circuit 102 is used with memory that is rewritable, such as RAM, SDRAM, and EEPROM, or a combination of such rewriteable memory.

[020] Digital data ready for encryption is stored in the data buffer 106. The digital data is moved from the data buffer 106 through the encryption circuit 102 to the I/O register 108 under the control of memory controller 104. The digital data is encrypted by the encryption circuit 102 prior to entering the I/O register 108. From the I/O register 108, the encrypted data is written to a memory location by the memory controller 104.

[021] The control block 118 of the encryption circuit 102 generates and maintains the encryption keys and sub-keys used in encrypting the digital data. The data I/O block 116 of the encryption circuit 102 processes the digital data going to or retrieved from memory.

[022] For decryption, the reverse process occurs and the encrypted digital data is decrypted as it is transferred from memory to the I/O register 108 for use via the data buffer 106 by the encryption circuit 102.

[023] In FIG. 2, a block diagram of the encryption circuit 102 FIG. 1 is shown. A buffer 202 contains the row/bank address data 216 from the memory controller 104. The key store 204 contains the encryption keys that are generated for each bank of memory. A multiplexer 206 is controlled by the row/bank address data contained in buffer 202 and selects an encryption key from the key store 204. The encryption key selected and the row/bank information are computed by a combiner 208 to form a sub-key.

[024] The sub-key is combined with digital data 228 that is sent through the data mixer 212 by combiner 210. The data mixer 212 scrambles portion of the memory data bus by rearranging the bits of each byte of digital data. Each reordered bit stays in its original byte lane to facilitate byte length memory writing. The output of combiner 210 is then transferred to the I/O register 108 and the signal 218 is made available to the control pad 112. In the exemplar implementation, there are four keys (key 0 220, key 1 222, key 2 224, key 3 226). In other implementations, other number of keys may be employed.

[025] Turning to FIG. 3, a block diagram of the encryption key generator 300 and key store 204 of FIG. 2 is shown. A reset signal 302, trigger signal 303 and enable signal 304 are inputs to an OR gate 306. The output 315 of the OR gate 306 is connected to an internal clock 308. The internal clock 308, in the exemplar implantation, is an oscillator that drifts and that may be affected by temperature and voltage. The internal clock 308 generates a clocking signal 312 that is received by the linear feedback shift register 310 and the key store 204.

[026] The linear feedback shift register 310 has a predetermined polynomial that may be assigned based on the customer, type of device or randomly. The predetermined polynomial may be programmed with fuses within an integrated circuit, PROMs, EPROMS, EEPROMS, or hardwired. The linear feedback shift register 310 generates a pseudorandom bit pattern 313.

[027] The bit pattern 313 is also received at the key store 204. While the enable signal 304 is received at the key store 204, no data is retained in the keys 220, 222, 224, and 226. When the enable signal 304 is removed, the bits from the polynomial in the key store are retained and made available to the encryption circuit 102.

[028] A random number output 314 is also made available from the linear feedback shift register 310 and is activated by the reset signal 302, trigger signal 303, or the enable signal 304. Whenever the trigger signal is present, a new random number 314 is made available by the linear feedback shift register 310 and internal clock 308. The random number 314 may be used by software within the digital device. For example, in an authentication or secure identification procedure. Additionally, when the reset signal 302 or enable signal 304 is present, a new random number 314 is generated by the linear feedback shift register 310 and internal clock 308. In other implementations, a random number may be held in the key store and regenerated whenever the keys 220, 222, 224, and 226 are regenerated.

[029] Turning now to FIG 4, a circuit diagram of the internal clock 308 of FIG. 3. The signal 315 from the OR gate 306 is received by the AND gate 402 along with a signal from the XNOR gate 404. When both signals are present, the output of the AND gate 402 is sent to a toggle flip-flop 406. The output of the toggle flip-flop 406 is fed back into the toggle flip-flop 406, the input of the XNOR gate 404, the follower flip-flop 408, and the input of XOR gate 410. The output of XOR gate 410 is provided to another input of the follower flip-flop 408. The

output of the follower flip-flop 408 is sent to the input of the XNOR gate 404, the input of the XOR gate 410 and the input of divider flip-flop 412. The output of the flip-flop 412 is feedback to the input of the divider flip-flop 412 in addition to being provided to the key store 204 as the clock signal 414.

[030] In FIG. 5, a circuit 500 with a zero primer circuit 502 employed with the linear feedback shift register 310 of FIG. 3 is shown. The linear feedback shift register 310 may power-on in any state but all zeros. If zero is the output in response powering the circuit, then the circuit will feedback zero and output zero again. In order to prevent the linear feedback shift register 310 from failing with all zeros, a zero primer circuit 502 is employed. A value of zero is hard coded as an input to the zero primer 502. When a zero value is received as an input 504 at the zero primer 502 (zero value is also output 510 by the linear feedback shift register 310) a one bit is forced into the linear feedback shift register 310 and generation of a bit stream continues.

[031] Turning to FIG. 6, a flow diagram 600 of the process steps for digital data encryption of digital data entering memory is shown. The process starts (602) when the digital device 100 is powered on or started (604). A linear feedback shift register 310 generates a pseudorandom bit pattern. When the digital device finishes starting (commonly called booting), the linear feedback shift register stops and the key store 204 stores the bit stream (606).

[032] A memory access command with bank and row information from the memory controller 104 initiates the sub-key generation (610). The bank selects a key from the key store and then the row forms a memory unit, such as a 32-bit word by replicating the byte. The 32-bit word and the selected key are combined resulting in a sub-key. The benefit of using a sub-key

is to mask the key if the rewritable memory in the digital device 100 has previously been previously set to known values, such as all zeros.

[033] Next, the data mixer 212 scrambles portions of the memory data bus by rearranging the bits of each byte (612). Each reordered bit must stay in its original byte lane since writes to memory may occur with whole bytes or groups of bytes. The next step is to combine the mixed data and sub-key (614). Each row of memory is a page in memory and each page has a unique sub-key and is exclusive "OR"ed with the digital data. The encrypted digital data is then stored in memory 616.

[034] If the digital device is reset or the memory system of the digital device is reset (618) then the linear feedback shift register (310) generates another pseudorandom bit pattern (606) and the key generation and encryption process is repeated as described above. Otherwise, the process is just waits unit the digital device is reset (618).

[035] Turning to FIG. 7, a flow diagram 700 of the steps of retrieving and decryption of digital data contained in memory is shown. The process starts (702) with the digital data has previously been encrypted and stored in the rewritable memory of digital device 100 (704). The memory controller 104 issues a memory access command that contains a bank and row (706). The encrypted digital data in rewriteable memory is accessed and retrieved (708). The sub-key is identical to the sub-key employed during encryption. The data mixer 212 unscrambles portions of the memory data bus by rearranging the bits of each byte (710). The unencrypted digital data is then made available to the data buffer 106 (712) and the process is complete (714).

[036] The foregoing description of an implementation has been presented for purposes of illustration and description. It is not exhaustive and does not limit the claimed inventions to the precise form disclosed. Modifications and variations are possible in light of the above

description or may be acquired from practicing the invention. For example, the described implementation uses hardware alone but the invention may be implemented as a combination of hardware and software or in software alone. Note also that the implementation may vary between systems. The claims and their equivalents define the scope of the invention.